The UltraLight Project: The Network as an Integrated and Managed Resource in Grid Systems for High Energy Physics and Data Intensive Science

Harvey Newman[1], Julian Bunn[1], Richard Cavanaugh[2], Iosif Legrand[1], Steven Low[1], Shawn McKee[3], Dan Nae[1], Sylvain Ravot[1], Conrad Steenberg[1], Xun Su[1], Michael Thomas[1], Frank van Lingen[1], Yang Xia[1]

[1]*California Institute of Technology, United States*
{newman,conrad,xsu,thomas}@hep.caltech.edu
{ julian.bunn, slow, fvlingen,yxia}@caltech.edu
{iosif.legrand,dan.nae,sylvain.ravot}@cern.ch

[2]*University of Florida*
cavanaug@phys.ufl.edu

[3]*University of Michigan*
smckee@umich.edu

Keywords: Network Protocols, Distributed Systems, Network Monitoring, Physics

Abstract

We describe the NSF-funded UltraLight project. The project's goal is to meet the data-intensive computing challenges of the next generation of particle physics experiments with a comprehensive, network-focused agenda. In particular we argue that instead of treating the network traditionally, as a static, unchanging and unmanaged set of inter-computer links, we instead will use it as a dynamic, configurable, and closely monitored resource, managed end-to-end, to construct a next-generation global system able to meet the data processing, distribution, access and analysis needs of the high energy physics (HEP) community. While the initial UltraLight implementation and services architecture is being developed to serve HEP, we expect many of UltraLight's developments in the areas of networking, monitoring, management, and collaborative research, to be applicable to many fields of data intensive e-science. In this paper we give an overview of, and motivation for the UltraLight project, and provide early results within different working areas of the project.

Introduction

The HEP community engaged in CERN's Large Hadron Collider (LHC) is preparing to conduct a new round of experiments to probe the fundamental nature of matter and space-time, and to understand the composition and early history of the universe. The decade-long construction phase of the accelerator and associated experiments (CMS[1], ATLAS[2], ALICE[3] and LHCb[4]) is now approaching completion, and the design and development of the computing facilities and software

[1] CMS: Compact Muon Solenoid (http://cmsinfo.cern.ch/)
[2] ATLAS: A Toroidal LHC ApparatuS (http://atlas.web.cern.ch/Atlas/)
[3] ALICE: A Large Ion Collider Experiment (http://alice.web.cern.ch/Alice/)
[4] LHCb: Large Hadron Collider beauty (http://lhcb.web.cern.ch/lhcb/)

is well-underway. The LHC is expected to begin operations in 2007. The experiments face unprecedented engineering challenges due to the volume and complexity of the experimental data, and the need for collaboration among scientists located around the world. The massive, globally distributed datasets which will be acquired, processed, distributed and analyzed are expected to grow to the 100 Petabyte level and beyond by 2010. Distribution of these datasets will require network speeds of around 10-100 gigabits per second (Gbps) and above. The data volumes are expected to rise to the Exabyte range, and the corresponding network throughputs to the 100 Gbps – 1 Terabit/sec range, by approximately 2015. In response to these challenges, the Grid-based infrastructures developed by collaborations in the US, Europe and Asia such as EGEE [5], OSG [6] and Grid3 [7] provide massive computing and storage resources. However, *efficient* use of these resources is hampered by the treatment of the interconnecting network as an external, passive, and largely unmanaged resource.

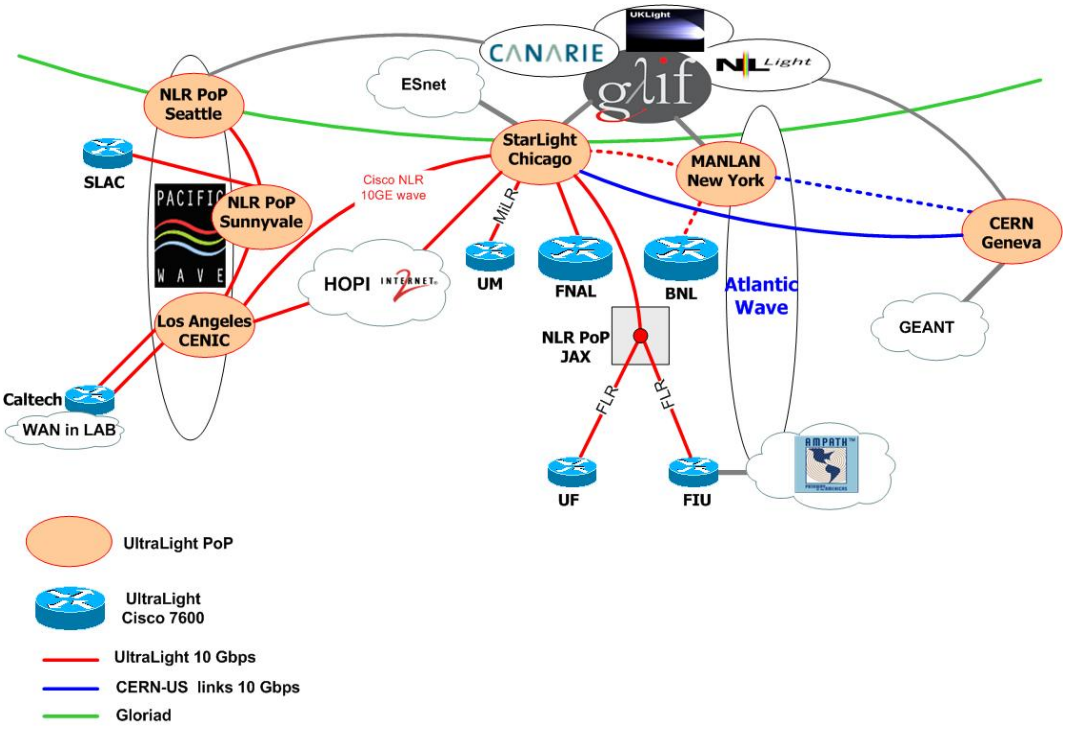

Figure 1. A schematic view of the initial UltraLight setup and connections to other major networks and WAN in Lab.

The UltraLight consortium of major HEP centers in the US was formed with the goal of addressing this deficiency. The project, funded by the National Science Foundation (NSF), is based on a strong partnership between the California Institute of Technology (Caltech), the University of Florida (UFL), the Florida International University (FIU), the University of Michigan (UM), the Stanford Linear Accelerator Center (SLAC), the Fermi National Accelerator Laboratory (FNAL), the Brookhaven National Laboratory (BNL), CERN, Universidade do Estado do Rio de Janeiro (UERJ) and the University of Sao Paulo (USP). The global capabilities and extent of the networking resources in UltraLight are shown in Figure 1, and include the major

[5] EGEE: Enabling Grids for E-sciencE (http://egee-intranet.web.cern.ch/egee-intranet/gateway.html)
[6] OSG: Open Science Grid (http://www.opensciencegrid.org/)
[7] Grid3: http://www.ivdgl.org/grid2003/

facilities of LHCNet[8] ,(the transatlantic 10 Gb/s backbone connecting CERN to the US in Chicago) , transcontinental 10 Gbps wavelengths from National Lambda Rail[9] (a major initiative of U.S. research universities and private sector technology companies to provide a national scale infrastructure for research and experimentation in networking technologies and applications) in partnership with Cisco Systems[10] , and Internet2's Abilene network[11], and partnerships with StarLight (a high-performance network exchange for many worldwide research and educational wide-area networks). Additional trans- and intercontinental wavelengths of our partner projects TransLight[12], Netherlight[13], UKlight[14], AMPATH[15], and CA*Net4[16] are used for network experiments on a part-time or scheduled basis.

We view UltraLight as being viable for e-Science because of the relatively limited scope of our virtual organizations (distributed data analysis). The advances we are proposing in its current form will most likely not scale to the general internet community. We are aware of scalability as an issue and are targeting our developments towards a large but manageable set of resources and VOs (~20).

The Network as a Resource

Within HEP there are typically two scenarios that make extensive use of the wide area network. In the first scenario, (raw) data from the detector located at CERN (Tier 0) produces so-called raw data at a rate of Petabytes per year, which will be processed locally. This reconstructed data will be stored at CERN and distributed in part to the Tier 1 centers around the world which in turn make it available to Tier 2 centers. The second scenario is related to analysis of the data in scenario 1. Physics analysis represents a "needle in the haystack" problem where physicists will analyze large datasets and identify in an iterative manner the datasets needed for physics discovery. Hundreds of physicists will perform various types of analysis at any time, using data that may potentially be distributed over several sites. During this process certain datasets become very populair or "hot" while other datasets languish or become "cold". In time data can change from "hot" to "cold", or vice versa depending on what data physicists are interested in. It is important that data will be readily available for physicists at the location where their jobs will be executed, which may require the typically large datasets, ranging from hundreds of Gigabytes to several Terabytes, to be replicated. This (potentially real time) replication combined with scenario 1 can overwhelm the network infrastructure and create a network "traffic jam" which in turn, can limit the use of computing resources as people can not get to their data.

The notion of treating the network as a managed resource is motivated by the general assumption that resources within a Grid (CPU, storage, network,…) will always be insufficient to meet the demand. This assumption is based on years of prior experience with computing in HEP, and it has design implications on the system as a whole. Such a resource constrained system requires policies that enforce "fair"-sharing . Fair in this context requires agreement on the part of the groups of users, or so-called virtual organizations (VOs) involved, on the terms under which they may use the available resources. Moreover, scheduling decisions need to take into account these

[8] http://www.datatag.org
[9] http://www.nlr.net/
[10] http://www.cisco.com/
[11] http://abilene.internet2.edu/
[12] http://www.startap.net/translight/
[13] http://www.surfnet.nl/innovatie/netherlight/
[14] http://www.uklight.ac.uk/
[15] http://www.ampath.fiu.edu/
[16] http://www.canarie.ca/canet4/

policies, together with the desired turnaround time profiles for each of several classes of work. To properly schedule and manage the system and to adhere to these policies, there needs to be detailed planning, monitoring and enforcement procedures that take the relevant information into account.

To achieve the goal of treating the network as an actively managed resource UltraLight focuses on four areas:

1. End-to-end monitoring, provides components with real time status information of the system as a whole, or of selected components. Monitoring information can be used by autonomous components to take decisions on the users' behalf or to optimize the system as a whole (e.g. optimize data throughput, CPU utilization, etc.).
2. Development and deployment of fundamental network and transfer protocols and tools such as FAST TCP [1],[2] and bandwidth and routing management technologies such as MPLS (MultiProtocol Label Switching) and QoS (Quality of Service) [3]
3. Deployment, testing and utilization of the UltraLight testbed and WAN in Lab (the "network wind tunnel")
4. (Physics) Application Level Services, that provide interfaces and functionalities for physics applications to effectively interact with the networking, storage and computation resources while performing (physics) analysis.

Results and applications developed in these four areas will be combined to provide an end-to-end service based system that supports Grid enabled physics analysis. The system will utilize the network as an active managed resource, will support thousands of users, will exploit the resources on the Grid, so as to allow the analysis of Petabytes of data, and contribute to active global collaboration on the discovery of new physics.

The UltraLight findings will be disseminated through the OSG. Indeed, several applications and frameworks discussed in the remainder of this paper have already become part of the OSG software distribution and are being deployed on the OSG testbed.

End-to-End Monitoring

Crucial to exposing the network as a managed resource, is the use of *end-to-end* monitoring which will permit applications and higher-level service layers to take into account increasingly advanced and complex behavior of the system as a whole. A new class of pro-active and reactive applications can be created that dynamically adapt to new and unforeseen system behavior. For example, network congestion or hardware component failures could be gracefully accommodated, and the availability of new network routes or capabilities could be exploited. These reactive applications will enhance the global system's resilience to malfunction, and will allow the optimization of resource usage, thereby enhancing both the overall task throughput and the effective implementation of policies, resulting in an increase in the speed with which physics results are obtained. This is the ultimate goal of the UltraLight effort.

End-to-end monitoring is thus vital within the UltraLight project to facilitate optimization of network resources. Part of the UltraLight planning is to implement a new set of global end-to-end managed monitoring services, building on the ongoing and rapidly advancing work with the MonALISA agent-based system [4].

MonALISA can be used to monitor and control network devices, such as routers and photonic switches. Since it gathers information system-wide, MonALISA is able to generate global views of the prevailing network connectivity, to identify network or end-system problems and act on

them strategically, or locally as required. Services that take decisions based on these (global) system views can be created and deployed: for example, mobile agents that are able to provide optimized dynamic routing for distributed applications have recently been added to MonALISA. At the time of writing, the MonALISA system includes 180 station servers (split between Grid and VRVS[17] [5] sites), around 10,000 participating nodes using over 60 WAN links, and monitoring of approximately 180,000 different operational parameters.

A key feature in MonALISA is the ability of independent processes, to publish and subscribe to data of other processes in the globally deployed system. Using a low level predicate mechanism within MonALISA, it is possible to create filters in these processes and associate these filters with certain actions. The combination of filters and associated actions can be viewed as a (rudimentary) form of policy specification.

As an example consider a process that is subscribed to bandwidth utilization of a network link and identifies (by filtering the subscribed data) a data movement activity that has been ranked "high priority". The associated action can then "throttle" the bandwidth usage for other processes, through "dynamically sized virtual pipes" by utilizing MPLS and QoS, improving the throughput of the high-priority transfer. This would allow a new higher priority task to be assigned the bandwidth it needs implicitly at the expense of less critical traffic. It is foreseen that each virtual organization should establish a set of policies to govern the priority rankings of different actions. In another instance MonALISA components could detect that a network link is saturated and re route additional traffic through other links. Using the "traffic" paradigm mentioned above, MonALISA would supply the "traffic lights" and steer dynamic "express lanes" in the network.

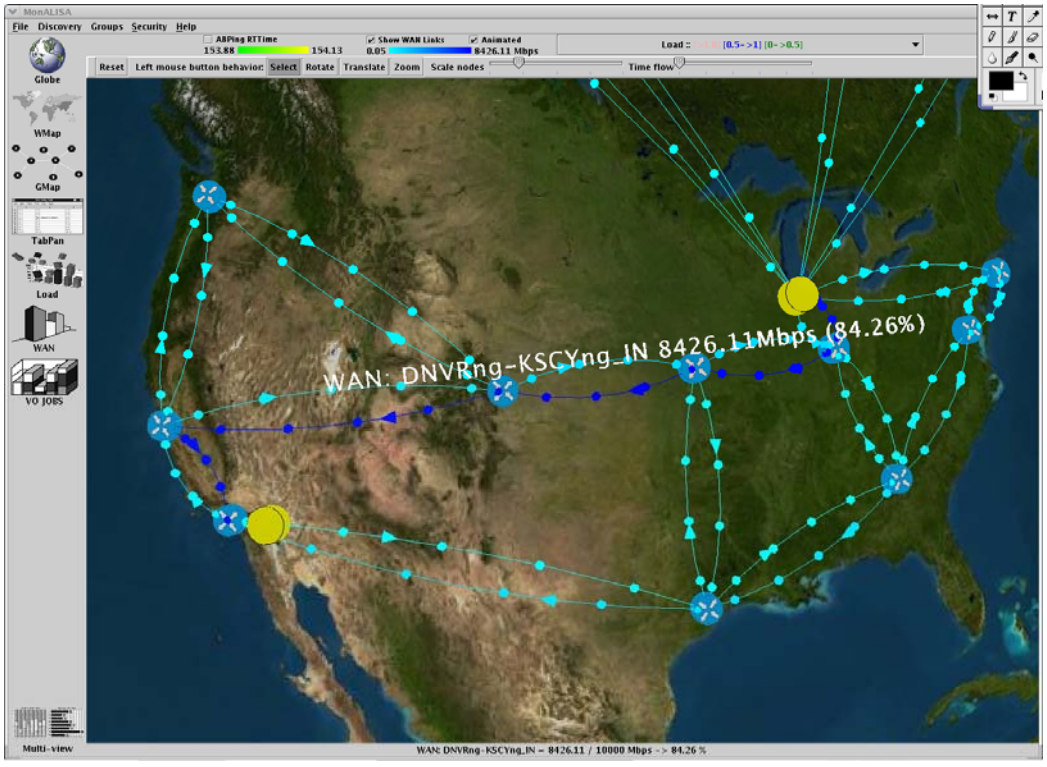

Figure 2. The MonALISA framework provides a distributed monitoring service system. Each MonALISA "station server" hosts and is able to schedule many dynamic services, and thereby acts as

[17] VRVS: Virtual Room Video conferencing System

a dynamic service system that can be discovered and used by any other services or clients that require such information.

A first example in end-to-end monitoring of resources has been the integration of MonALISA and the VRVS. MonALISA was adapted and deployed on VRVS reflectors, to collect information about the topology of the VRVS system, to monitor and track traffic among reflectors and report communication errors with the peers, and to track the number of clients and active virtual rooms. In addition overall system information is monitored and reported in real time for each reflector, such as CPU usage and total traffic in/out. Agents within MonALISA have been developed to provide and optimize dynamic routing of the VRVS data streams. These agents require information about the quality of alternative connections and solve a minimum spanning tree problem to optimize data flow at the global level. The latter is also important within the UltraLight project where you want to optimize data flows too.

Figure 2 shows the MonALISA system in action during SC2004 [18]. MonALISA gathers arbitrarily complex monitoring information in the global system and processes it in its distributed agent framework. The framework scales well with system size due to the use of agents, a multithreaded engine that hosts a variety of loosely-coupled and self-describing dynamic services, and the ability of each service to register itself and then be "discovered" and made use of by other services or clients.

MonALISA has been used in several Grid projects, such as Grid3, is part of OSG and monitors several of the major networks such as Abelene and Gloriad. One of the reasons for using MonALISA within the UltraLight project is that it offers (amongst other) features such as: global scalability, low level policy specifications, autonomous publish/subscribe functionality, and the ability to steer other applications based on monitor information. These features are not (or only partly) available in other Grid based monitoring systems such as Ganglia [6], R-GMA [7], and GridICE [8].

Protocols and Tools

FAST TCP is an implementation of TCP with a new congestion control algorithm that is optimized for high speed long distance transfers. While the congestion control algorithm in the current TCP implementation uses packet loss as a measure of congestion, FAST TCP uses round-trip delay (time from sending a packet to receiving its acknowledgment). This allows FAST TCP to stabilize at a steady throughput without having to perpetually push the queue to overflow as loss-based schemes inevitably do. Moreover, delay has the right scaling with link capacity that enhances stability as networks scale up in capacity and geographical size[9].

In addition to many experimental evaluations of FAST TCP in real networks and emulated testbeds, we have also modeled it mathematically and analyzed its equilibrium and stability properties. In equilibrium, FAST TCP allocates bandwidth among competing flows in general networks according to proportional fairness. It is a fairness notion that favors small flows but less extremely than maxmin fairness [10]. Moreover the equilibrium point always exists and is unique for an arbitrary network. The equilibrium point has also been proven to be stable under various assumptions [9]. Figure 3 shows a comparison between FAST TCP and RENO TCP, where the latter is based on the RENO fast retransmit algorithm [11].

[18] http://www.sc-conference.org/sc2004/

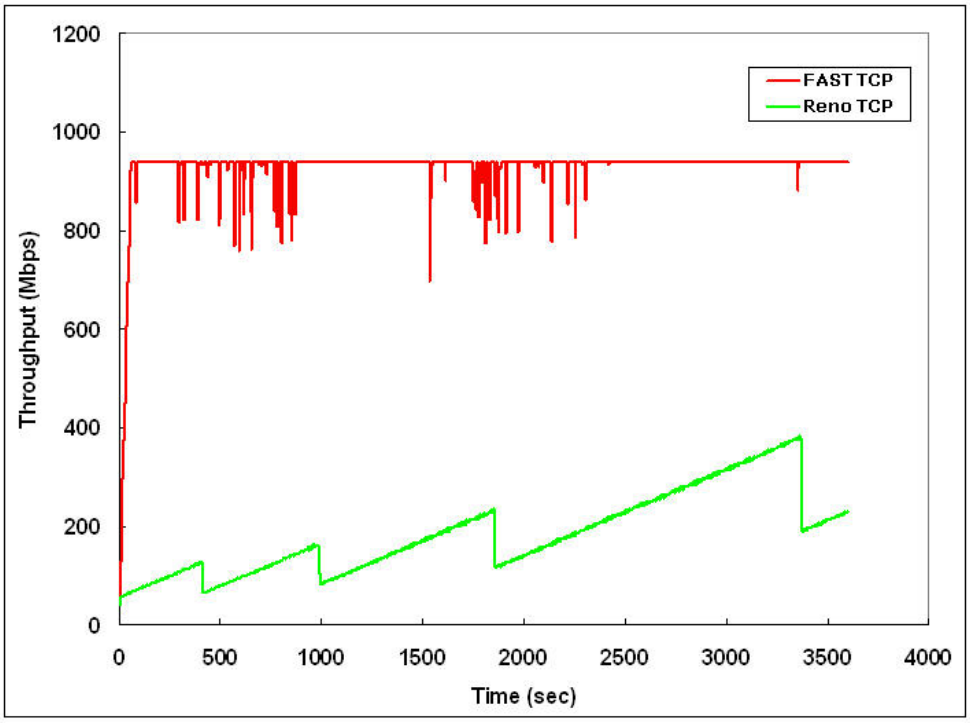

Figure 3. The throughput of FAST flows compared with RENO, in the presence of packet loss.

UltraLight intends to exploit recent advances in robust control theory and convex optimization to guide the operation of the large scale network we build. Recently developed techniques at Caltech, use sum of squares optimization methods to provide convex polynomial time-relaxations for many NP-hard problems involving positive polynomials. The observation that sum of squares decompositions of multivariate polynomials can be computed efficiently using semi-definite programming has initiated the development of software tools, such as SOSTOOLS [12], that facilitate the formulation of the semi-definite programs from their sum of squares equivalents UltraLight will develop methods of incorporating SOSTOOLS into the MonALISA monitoring and control software system. This will allow the calculation of stability regions for a given network operation regime, and the derivation of control actions that will steer the network so as to remain within a desired performance regime.

UltraLight High Speed Networking

The UltraLight hybrid packet- and circuit-switched network infrastructure employs both "ultrascale" protocols such as FAST, and the dynamic creation of optical paths for efficient fair sharing on long range networks in the 10 Gbps range. Recently, UltraLight project members, led by Caltech, broke the Internet2 Land Speed Record[19] by sending 2.9 Terabytes of data across 26,950 kilometers of network in one hour using Caltech's FAST TCP [1] , at an average rate of 6.86 Gigabits per second (Gbps). The same week, the team captured the Supercomputing Bandwidth Challenge award[20] for sustained bandwidth at SC2004, by generating an aggregate rate of 101 Gbps to and from the show floor at Pittsburgh; the same level of data throughput expected to occur in the early phases of operation of the LHC experiments.

[19] http://lsr.internet2.edu
[20] http://pr.caltech.edu/media/Press_Releases/PR12620.html

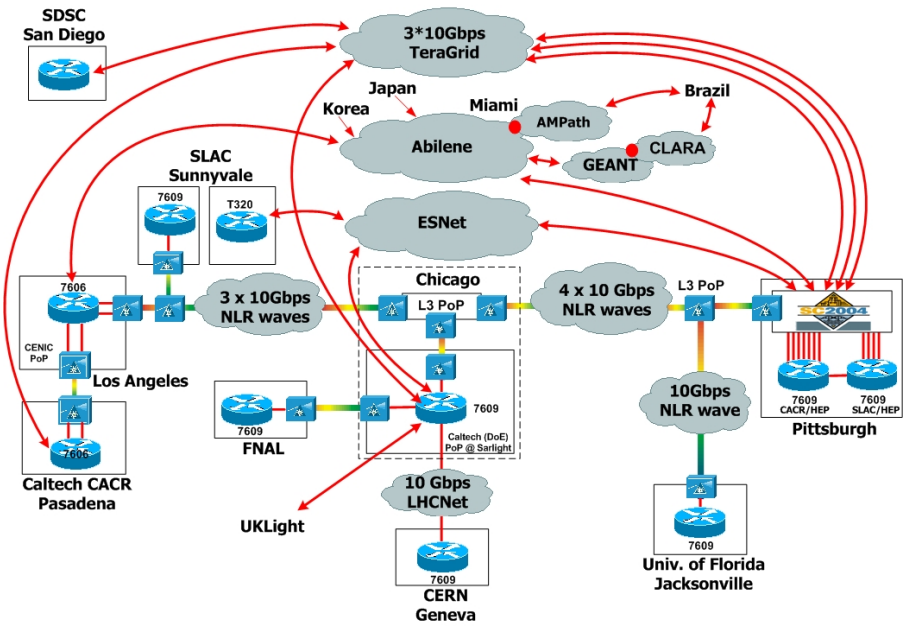

Figure 4. High Speed TeraByte Transfers for Physics Demonstration (SC2004)

The achieved bandwidth was made possible in part through the use of the FAST TCP protocol, through the use of seven 10 Gbps links to the Caltech Center for Advanced Computing (CACR) booth and three 10 Gbps links to the SLAC/Fermilab booth, and through the combined use of wide area links provided by National Lambda Rail, Internet2's Abilene, LHCNet, the Energy Sciences Network (ESNet)[21], the TeraGrid[22], links to Brazil over the AMPATH, CHEPREO[23] link, and via and EU-funded CLARA link[24] and the GEANT[25] pan-European network, and a link to Korea as shown in Figure 4. Other key elements in the diagram include two 10 Gbps links between Caltech and Los Angeles provided by Cisco Systems, connections to the Jacksonville location on Florida Lambda Rail (FLR), and a 10 Gbps dark-fiber link between Starlight and Fermilab. Apart from the connections in Pittsburgh, most of the network connections shown in the figure persist as part of the UltraLight network testbed today.

The aim of these bandwidth challenges is to show that given the combination of hardware and software it is possible to create a "super highway" for data transfer that can meet the challenges of the physics community. A next step in the bandwidth challenges will be disk-to-disk transfers and the ability to analyze potentially remote data as if it where on a local disk. Although this type of analysis will not be available to all the users in the system, it will be possible to make it available to high priority users and processes, such that we can offer a more agile system in which we offer the flexibility to assign resources (CPU, storage and network) to promising physics analysis processes or groups, which would lead to new scientific discoveries faster.

WAN in Lab

The Bandwidth Challenge at SC2004 mentioned earlier provided an ideal environment for dedicated access to network resources for a limited amount of time. Although monitoring

[21] http://www.es.net/
[22] http://www.teragrid.org/
[23] http://www.chepreo.org/
[24] http://www.redclara.net/
[25] http://www.geant.net/

applications were deployed on many of the network resources we had no control over others which made it difficult to diagnose low level failures of components. In order to analyze every aspect of the network and applications interacting with network resources, the UltraLight consortium will make use of an unusual facility: WAN in Lab.

WAN in Lab is a unique testbed that is being built at Caltech. It is funded by NSF, ARO, Cisco and Caltech. WAN in Lab[26] is literally a wide-area-network – it includes 24,000 kilometers of fibers, optical amplifiers, dispersion compensation modules, WDM (Wavelength Division Multiplexing) gear, optical switches, routers, and servers - but it is housed in a single laboratory at Caltech! The initial hardware, anticipated to be operational in the Fall of 2005, will have 6 Cisco ONS 15454 switches, 4 Cisco 7609 routers, and a few dozen high speed servers. We intend to connect it by 10Gbps link to, and make it an integral part of, UltraLight (Figure 1). This extends the round-trip time of an end-to-end connection between a server in WAN in Lab and one in a global production network to more than 300ms. The 300ms number is important because it is larger but of the same scale as the largest round-trip times we expect in the "real" networks. This allows us to insure our work is relevant for global networks. We also intend to connect WAN in Lab to the Sunnyvale and Seattle GigaPoPs (Figure 5).

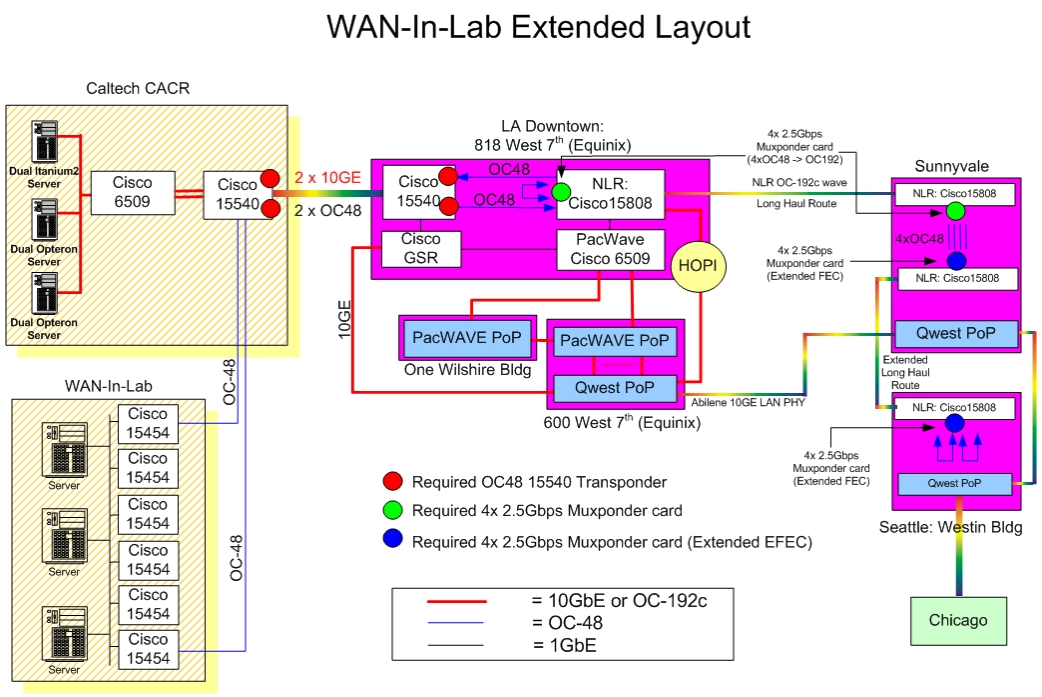

Figure 5. WAN in Lab extended layout

WAN in Lab offers a unique environment for developing and testing protocols for sharing resources across high speed WANs, and complements synergistically with the UltraLight infrastructure. It is the “wind tunnel of networking research”. Distributed systems tools by and for the UltraLight community can be developed, debugged, and tested first on WAN in Lab before they are progressively deployed on UltraLight. This will greatly shorten the cycle of design, development, testing, and deployment for the community.

[26] http://netlab.caltech.edu/WiL/

Application Level Services

Ultimately the network as a managed resource needs to be integrated into existing globally distributed systems. The Grid Analysis Environment (GAE) [13] describes an application level Service Oriented Architecture (SOA) to support end-to-end (physics) analysis. It describes the ensemble of services (and there interactions) such as discovery [14], scheduling [15], submission [16], job tracking, that will be exposed to users and domain applications. UltraLight is extending the GAE to the UltraLight Analysis Environment (UAE). The UAE focuses on integration between components identified in the GAE and components that expose the network as a managed resource.

Most of the users will utilize Grid resources through Grid portals that hide much of the (Web Service) infrastructure and resource complexity of the Grid. Components of the GAE will interact with monitor applications, replicate data, schedule jobs, and find optimal network connections in an autonomous manner that would result in a self organizing Grid that minimizes single point of failures, in which thousands of users are able to get fair access to a limited set of distributed resources of the Grid in a responsive manner. Many of the Web Service implementations within the UAE will be made available through and developed in Clarens [14], a Web Service based framework, such as Globus[17] and Glite [18], available in Python and Java implementations, that offers several additional features: X.509 Certificate based authentication when establishing a connection, access control on Web Services, remote file access (and access control on files), discovery of services and software, virtual organization management, high performance (measured 1400 calls/second), role management (role management is being extended to interoperate with GUMS servers [19]), and support for multiple protocols (XML-RPC, SOAP, Java RMI, JSON-RPC). Integration between the network resources and Web Services will be achieved through MonALISA [14].

At SC2003 and 2004 an early GAE prototype was demonstrated that enabled users to submit jobs through the SPHINX Grid scheduler [15] to a site. SPHINX differs from other schedulers such as Pegasus [20] in that is supports a decentralized policy based environment for scheduling. The scheduling decisions where based on MonALISA monitoring data which consisted amongst others of the sites queue length, and the speed at which a job finished once it started running. SPHINX then actively monitored the progress of the job while in the queue and if necessary rescheduled it to another site. An extension to this prototype provided a wrapper around the job which sent back job state information to MonALISA and BOSS[27] [16]. BOSS stored very detailed information on the jobs states, including error logs, while MonALISA stored real time information regarding the state of the job e.g. submitted, started, running, finished, crashed,…..). This information was then visualized by plotting state against time in a so-called life line plot.

More recently a software and service discovery Web Service has been implemented as Clarens Web Services. Both services utilize MonALISA to disseminate discovery information. The services provide a dynamic real time view of what Web Services and software applications are available within the distributed system.

The SPHINX scheduler and the discovery services are two early examples of integration of application level services with an end–to-end monitor system, to provide a global view of the system. The discovery services are part of the OSG and have been deployed on the OSG testbed.

[27] Batch Object Submission System: provides wrappers for jobs for monitoring detailed job status.

Recent work has begun on creating a reservation service in collaboration with the Lambda Station project[28] on a service that allocates bandwidth (where that is possible) in multiple network domains from a source to a target, at a time a user requests it. Such a service is important for Grid operators who are in charge of moving large amounts of data through multiple network domains. In most cases this data movement does not have to happen immediately but can be planned. In the near future manual operation of these data transfers can be replaced with autonomous applications (e.g. agents) that monitor data access and decide to reserve bandwidth for data transfer.

Summary

The UltraLight project marks the entry into a new era of global real time responsive systems where all three sets of resources - computational, storage and network - are monitored and tracked to provide efficient, policy-based resource usage, and optimized distributed system performance on a global scale. In addition to a network testbed of unprecedented scope, both in the field and in the laboratory (Wan in Lab), UltraLight relies on sophisticated applications built on top of advanced network protocols such as FAST, and autonomous service-oriented frameworks such as MonALISA and Clarens. By consolidating with other emerging data-intensive Grid systems, UltraLight will drive the next generation of Grid developments, and support new modes of collaborative work. Such globally distributed systems will serve future advanced applications in many disciplines, bringing great benefit to science and society. UltraLight paves the way for more flexible, efficient sharing of data by scientists in many countries, and could be a key factor enabling the next round of discoveries at the HEP frontier, soon to be explored at the LHC. Advancements in UltraLight, through Caltech's VRVS system, also could have profound implications for integrating information sharing and on-demand audiovisual collaboration in our daily lives, with a scale and quality previously unimaginable. Pro actively managing the (wide area) network as a (limited) resource to enable fair usage is a novel approach within Grid projects. At the time of writing this article, projects such as EGEE and the OSG view the network as a largely unmanaged resource that will be (passively) monitored. EGEE has a network resource provision activity that identifies scenarios [21] which will benefit from the work being undertaken in the UltraLight project. The OSG network activity will be driven by the UltraLight community, as members of this community are also active members within the OSG. Applications such as MonALISA and the Clarens service and software discovery service are part of the OSG software releases and provide the means to disseminate UltraLight results into the OSG community.

Acknowledgements

This work is partly supported by the Department of Energy grants: DE-FC02-01ER25459, DE-FG03-92-ER40701, DE-AC02-76CH03000 as part of the Particle Physics DataGrid project, DE-FG02-04ER-25613 as part of Lambda Station, and by the National Science Foundation grants: ANI-0230967, PHY-0218937, PHY-0122557, PHY-0427110, ANI-0113425, ANI-0230967, EIA-0303620, by the ARO grants: DAAD19-02-1-0283, F49620-03-1-0119 and the AFOSR grants: F49620-03-1-0119. We would also like to acknowledge the generous support of Cisco. Any opinions, findings, conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the Department of Energy or the National Science Foundation.

[28] http://www.lambdastation.org/

References

[1] Cheng Jin, David X. Wei and Steven H. Low. "*FAST TCP: motivation, architecture, algorithms, performance"*, Proceedings of the IEEE Infocom, Hong Kong, March 2004. (see also: http://netlab.caltech.edu/FAST)

[2] C. Jin, D. X. Wei, S. H. Low, G. Buhrmaster, J. Bunn, D. H. Choe, R. L. A. Cottrell, J. C. Doyle, W. Feng, O. Martin, H. Newman, F. Paganini, S. Ravot, S. Singh. "*FAST TCP: From Theory to Experiments*", IEEE Network, 19(1):4-11, January/February 2005.

[3] X. Xiao, L. M. Ni, "Internet QoS: A Big Picture", in IEEE Network, 13(2):8-18, March, 1999

[4] H.B. Newman, I.C. Legrand, P. Galvez, R. Voicu, C. Cirstoiu "*Monalisa : A Distributed Monitoring Service Architecture.*" In proceedings of Computing for High Energy Physics (CHEP), Paper ID: MOET001, La Jolla, California, June 2003.. (see also: http://monalisa.caltech.edu/)

[5] D. Adamczyk, G. Denis, J. Fernandes, P. Farkas, P. Galvez, D. Lattka, I. Legrand, H. Newman, J. Sucik, K. Wei,"*A Globally Distributed Real Time Infrastructure for World Wide Collaborations*", In proceedings of Computing for High Energy Physics (CHEP), Paper ID:88, Interlaken, Switzerland, September 2004.

[6] M. L. Massie, B. N. Chun, D.E. Culler, "*The Ganglia Distributed Monitoring System: Design, Implementation, and Experience*", Parallel Computing 30(7):817-840, July 2004.

[7] A. Cooke, A. Gray, L. Ma, et al. "*R-GMA: an Information Integration System for Grid Monitoring*", In proceedings of the 11th International Conference on Cooperative Information Systems (CoopIS 2003) pp 462-481, Catania, Italy, November 2003.

[8] S. Andreozzi, N. De Bortoli, S. Fantinel, A. Ghiselli, G. Rubini, G. Tortone, M. Vistoli, "*GridICE: a Monitoring Service for Grid Systems*", Preprint. to appear in Future Generation Computer Systems journal, Elsevier.

[9] J. Wang, D. X. Wei and S. H. Low. "*Modeling and stability of FAST TCP."* In proceedings of the IEEE Infocom, Miami, Florida,, March 2005.

[10] F. P. Kelly, A.K. Maulloo and D. K. H. Tan. "*Rate Control in Communication Networks: Shadow Prices, Proportional Fairness and stability*", Journal of the Operational Research Society 49 (1998), 237-252.

[11] W. Stevens, M. Allman, V. Paxson, *"TCP congestion Control"* RFC2581 , April 1999.

[12] Stephen Prajna, Antonis Papachristodoulou, Peter Seiler, and Pablo A. Parrilo. "*SOSTOOLS and its Control Applications. Positive Polynomials in Control*". D. Henrion and A. Garulli (Eds.). Springer-Verlag. 2005.

[13] F. van Lingen, J. Bunn, I. Legrand, H. Newman, C. Steenberg, M. Thomas, P. Avery, D. Bourilkov, R. Cavanaugh, L. Chitnis, M. Kulkarni, J. Uk In, A. Anjum, T. Azim "*Grid Enabled Analysis: Architecture, Prototype and Status*" in proceedings of Computing for High Energy Physics (CHEP) Interlaken, Switzerland September 2004. (see also: http://ultralight.caltech.edu/gaeweb/portal)

[14] F. van Lingen, J. Bunn, I. Legrand, H. Newman, C. Steenberg, M. Thomas, A. Anjum, T. Azim, "*The Clarens Web Service Framework for Distributed Scientific Analysis in Grid Projects"*, In proceedings of the International Conference on Parallel Processing pp 45-52, Oslo, Norway, June 14-17, 2005. (see also: http://clarens.sourceforge.net/)

[15] J. In, P. Avery, R. Cavanaugh, S. Ranka, "*Policy Based Scheduling for Simple Quality of Service in Grid Computing*", In Proceedings of 18th International Parallel and Distributed Processing Symposium, April 26 –30, Santa Fe, 2004.

[16] Grandi, C., Renzi, A., "*Object Based System for Batch Job Submission and Monitoring (BOSS)"*, CMS note 2003/005.

[17] Foster, C. Kesselman, "*Globus: A Metacomputing Infrastructure Toolkit*" Intl. J. Supercomputer Applications, 11(2):115-128, 1997.

[18] M. Lamanna, B. Koblitz, T. Chen, W. Ueng, J. Herrala, D. Liko, A. Maier, J. Moscicki, A. Peters, F. Orellana, V. Pose, A. Demichev, D. Feichtinger, "*Experiences with the gLite Grid Middleware*" In proceedings of CHEP, Interlaken Switzerland, 2004. (see also: http://glite.web.cern.ch/glite/)

[19] G. Carcassi, T. Carter, Z. Liu, G. Smith, J. Smith, J. Spiletic, T. Wlodek, D. Yu, X. Zhao, "*A Scalable Grid User Management System for Large Virtual Organizations*", In proceedings of Computing for High Energy Physics (CHEP), Interlaken, Switzerland, September 2004.

[20] E. Deelman, J. Blythe, Y. Gil, C. Kesselman, G. Mehta, S. Patil, M.-H. Su, K. Vahi, M. Livny, "*Pegasus : Mapping Scientific Workflows onto the Grid*", Across Grids Conference, Nicosia, Cyprus, 2004.

[21] EGEE scenarios: https://edms.cern.ch/document/476742